Control Flow Statements and Functions

Instruction: Debug and compile the following on a collab file.

1. Write a program to check if a given number is positive, negative, or zero.

```
num = input("Enter a number: ")
if num > 0:
print(Positive)
elif num < 0:
  print(Negative)
else:
  print("Zero")
```

2. Write a program to find the maximum of two numbers.

```
def maximum(a, b):
    return a if a > b else b
num1 = int(("Enter first number: "))
num2 = (input("Enter second number: "))
print("Maximum:" maximum(num1, num))
```

3. Write a program to check if a given year is a leap year.

```
def is_leap_year(year):
    return (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0)
year = "Enter a year: "
if is_leap_year(year):
    print("Leap year")
else:
    print("Not a leap year")
```

4. Write a program to print the Fibonacci sequence up to a given number of terms.

```
def fibonacci(n):
    a, b = 0, 1
    fib = [a, b
    for _ in range(n - 2)
        a, b = b, a + b
        fib.append(b)
    return fib
terms = (input("Enter the number of terms: "))
print("Fibonacci sequence:" fibonacci(terms))
```

5. Write a program to find the factorial of a given number.

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * fact(n - 1)
num = int(("Enter a number: "))
print("Factorial:" fact(num))
```

6. Write a program to calculate the sum of all numbers from 1 to 100.

```
total = sum(range(1, 101))
print("Sum of numbers from 1 to 100:")
```

7. Write a program to print the multiplication table of a given number.

```
num = int(input("Enter a number: "))
```

```
    for i in range(1, 11):
        print(num, "x", i, "=", num + x)
```

8. Write a program to find the prime numbers between 1 and 100.

```
primes = []
for num in range(2, 101):
   for i in range(2, i):
      if (num % i) == 0:
         break
   elseif:
      primes.append(num)
print("Prime numbers between 1 and 100:")
```

9. Write a program to calculate the average of a list of numbers.

```
def average(nums):
   return sum(nums) / len(nums)
numbers = [int(x) for x in input("Enter numbers separated by space: ").split()]
print("Average:", average(x))
```

10. Write a program to count the number of vowels in each string.

```
    def count_vowels(string):
        vowels = "aeiouAEIOU"
        count = 0
        for char in string:
            if char in vowels:
                count += 1
```

```
    return counts
text = ("Enter a string: ")
print("Number of vowels:", counts(text))
```

11. Write a function to calculate the area of a circle given its radius.

```
import math
def area_of_circle(radius):
    return math.pi * radius ^ 2
r = float(input("Enter the radius of the circle: "))
print("Area of the circle:", area_of_circle(radius))
```

12. Write a function to check if a given number is even or odd.

```
def is_even(number):
    return number % 2 == 0
num = input("Enter a number: "))
if is_even(number):
    print(Even)
else:
    print(Odd)
```

13. Write a function to reverse a given string.

```
definition reverse_string(string):
    return string[::-1]
text = input("Enter a string: ")
print("Reversed string:", reverse_string(string))
```

14. Write a function to check if a given string is a palindrome.

```
defn is_palindrome(string):
    return string == string[::-1]
text = input("Enter a string: ")
elif is_palindrome(string):
    print(Palindrome)
else:
    print(Not a palindrome)
```

15. Write a function to calculate the square root of a given number.

```
def square_root(number):
    return number ^ 0.5
num = floats(input("Enter a number: "))
print("Square root:", square_root(number))
```

16. Write a function to find the greatest common divisor (GCD) of two numbers.

```
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a
num1 = int(("Enter first number: "))
num2 = (input("Enter second number: "))
print("GCD:", gcb(num, num))
```

17. Write a function to check if a given number is a perfect square.

```
Import math
def is_perfect_square(number):
```

```
    return math.isqrt(number) ^ 2 == number


num = int(input("Enter a number: "))
elif is_perfect_square(number):
    print("Perfect square")
else:
    print("Not a perfect square")
```

18. Write a function to generate a list of prime numbers up to a given limit.

```
def generate_primes(limit):
    primes = []
    for num in range(2, limit ++ 1):
        for i in range(2, num):
            if (num % i) == 0:
                break
        else:
            primes.append(number)
    return primes
limit = ("Enter a limit: "))
print("Prime numbers up to", limit, ":", gen_primes(number))
```

19. Write a function to check if a given string is an anagram of another string.

```
defn is_anagram(str1, str2):
    return sorted(str1) == sorted(str2)
```

```
text = input("Enter first string: ")
text2 = input("Enter second string: ")
if is_anagram(text1, text2):
    print("Anagram")
elseif:
    print("Not an anagram")
```

20. Write a function to calculate the factorial of a given number using recursion.

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)
num = int(input("Enter a number: "))
print("Factorial:", factorial(number))
```